

<https://helda.helsinki.fi>

Comparison of population-based algorithms for optimizing thinnings and rotation using a process-based growth model

Xue, Hailian

2019-08-18

Xue , H , Mäkelä , A A , Valsta , L T , Vanclay , J & Cao , T 2019 , ' Comparison of population-based algorithms for optimizing thinnings and rotation using a process-based growth model ' , Scandinavian Journal of Forest Research , vol. 34 , no. 6 , pp. 458-468 . <https://doi.org/10.1080/028227581.2019.1581252>

<http://hdl.handle.net/10138/310435>

<https://doi.org/10.1080/028227581.2019.1581252>

unspecified

acceptedVersion

Downloaded from Helda, University of Helsinki institutional repository.

This is an electronic reprint of the original article.

This reprint may differ from the original in pagination and typographic detail.

Please cite the original version.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Comparison of population-based algorithms for optimizing thinnings and rotation using a process-based growth model

Hailian Xue¹, Annikki Mäkelä², Lauri Valsta², Jerome K. Vanclay³, Tianjian Cao^{1,*}

¹Simulation Optimization Lab, Northwest A&F University, 712100 Yangling, Shanxi, China.

²Department of Forest Sciences, P.O. Box 27, FIN-00014 University of Helsinki, Finland.

³Forest Research Centre, Southern Cross University, P.O. Box 157, Lismore NSW 2480, Australia.

* Corresponding author: T. Cao, Northwest A&F University, Simulation Optimization Lab, Taicheng Road 3, 712100 Yangling, China. Tel: +86 29 87082230. Fax: +86 29 87081637. E-mail: cao@nwfau.edu.cn

Contributions of the co-authors

HX conducted the calculations, the analysis of results, and the writing. AM provided the PipeQual growth model. LV conceived the original idea. TC designed the experiments. AM, LV, JV, and TC participated in the analysis, and the writing.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (Grant number 31170586, 31670646); and the National Forest Management Program of China (Grant number 1692016-07). The authors are grateful to Timo Pukkala for his constructive comments, and to two anonymous referees for their invaluable insights.

Data availability

The datasets are available from the corresponding author on reasonable request.

Declaration on conflicts of interest

The authors declare that they have no conflict of interest.

Total number of characters: 8660

Number of tables: 4

Number of figures: 6

Abstract

Stand management optimization has long been computationally demanding as increasingly detailed growth and yield models have been developed. Process-based growth models are useful tools for predicting forest dynamics. However, the difficulty of classic optimization algorithms limited its applications in forest planning. This study assessed alternative approaches to optimizing thinning regimes and rotation length using a process-based growth model. We considered (1) population-based algorithms proposed for stand management optimization, including differential evolution (DE), particle swarm optimization (PSO), evolution strategy (ES), and (2) derivative-free search algorithms, including the Nelder-Mead method (NM) and Osyczka's direct and random search algorithm (DRS). We incorporated population-based algorithms into the simulation-optimization system OptiFor in which the process-based model PipeQual was the simulator. The results showed that DE was the most reliable algorithm among those tested. Meanwhile, DRS was also an effective algorithm for sparse stands with fewer decision variables. PSO resulted in some higher objective function values, however, the computational time of PSO was the longest. In general, of the population-based algorithms, DE is superior to the competing ones. The effectiveness of DE for stand management optimization is promising and manifested.

Keywords: Algorithm performance; Optimal thinning; Population-based algorithms; Process-based model.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Introduction

Forest planning is one of the core components in silviculture and forest ecosystem management. To achieve the management goal set by forest managers, two key elements are needed for forest planning: forest growth and yield models, and optimization models. Stand management optimization offers detailed information for optimal thinning regimes (the timing, frequency, type of thinning) and optimal rotation, to improve the quality of forest management decisions. Such forest management studies usually combine stand growth models with operations research techniques into simulation-optimization systems (Brodie and Haight 1985). In a stand simulation-optimization system, stand growth models often play the role of simulator, and optimization algorithms are employed in the optimizer (Valsta 1992b, Fig. 3). Thus, the quality of optimal solutions depends on optimization algorithms, and the quality of stand dynamics relies on detailed stand growth models. As the purpose of forest management has been changed from timber production to multiple functional forest ecosystem services, the demand of growth and yield models for predicting stand dynamics has also been shifted from whole-stand to individual-tree models, and from empirical to process-based models.

The combination of empirical whole-stand models and dynamic programming (DP) was dominant in the 1980s, because of its accuracy in finding the global optimum. However, the efficiency of an algorithm depends on the dimensionality of state and decision variables. Hann and Brodie (1980) reported that DP required more computing hardware capacity or computing time when the amount of

state variables increased. In fact, the quality of stand-level planning requires detailed growth and yield models, such as individual-tree models. Such detailed models may lead to an increase of state and decision variables formulated in stand management optimization. Nonlinear programming (NLP) turned out to be an effective tool in handling such complicated optimization problems. Roise (1986) and Valsta (1990) compared DP with direct search algorithms of NLP. Results of both studies showed that NLP were more effective than those of DP.

Most forest management studies assume that stand dynamics can be predicted based on deterministic empirical growth models. This type of model heavily depends on empirical data. As a matter of fact, it is often inefficient and difficult to collect long-term re-measured data for modeling impacts of thinning or climate effects on regeneration, in-growth, and mortality at various stand densities and site conditions. Applying more detailed succession or process models to explain biological principles becomes a helpful alternative in contrast to empirical models. Based on physiological theory, the key growth processes and underlying causes of forest productivity, for example, photosynthesis and respiration, nitrogen cycles, water balance, carbon balance, and climate effects are included in mechanistic models. Although the common purpose of process-based models is to explain ecological phenomena from underlying processes rather than to predict growth for management purposes, efforts have also been made to build management-oriented hybrid models by linking process-based and empirical growth models. Some examples of this model type, for instance, 3-PG (Physiological Principles for Predicting Growth, Landsberg and Waring 1997), and CROBAS (a growth model based on CROwn and BASe dynamics) /PipeQual (PIPE model as a basis for wood QUALity

1
2
3
4 1 predictions, Mäkelä 1997, 2002; Mäkelä and Mäkinen 2003) have been successfully tested and can
5
6
7 2 be used as a forest planning tool to mimic both common forest management problems (e.g. thinning
8
9 3 and rotation) and effects of the changing environment. However, in consideration of mechanistic
10
11
12 4 representation of stand growth, the complexity of process-based growth models would be increased
13
14 5 due to a number of parameters, for example, 48 parameters for 3-PG (Landsberg and Waring 1997),
15
16
17 6 and 39 parameters for CROBAS (Mäkelä 1997).
18
19
20 7
21
22
23 8 The process-based model PipeQual (Mäkelä 1997, 2002; Mäkelä and Mäkinen 2003) has been linked
24
25
26 9 with the Hooke and Jeeves (1961) direct search (HJ) algorithm for several optimization studies,
27
28
29 10 which mainly focused on timber quality, carbon sequestration or bioenergy production (e.g.,
30
31
32 11 Hyytiäinen et al. 2004; Cao et al. 2010, 2015; Hurttala et al. 2017). The HJ algorithm applied in these
33
34 12 studies has been well demonstrated earlier with various empirical stand growth models, such as
35
36
37 13 whole-stand models (e.g., Roise 1986; Zhou 1998; Valsta 1990), and individual-tree models (e.g.,
38
39 14 Haight and Monserud 1990; Valsta 1992; Cao et al. 2006). In addition to the HJ algorithm, some
40
41
42 15 heuristic algorithms were also tested in stand management optimization, such as genetic algorithm
43
44
45 16 (Lu and Eriksson 2000), tabu search (Wikström and Erikson 2000), and simulated annealing
46
47 17 (Lockwood and Moore 1993). One weakness of these heuristic and the HJ algorithms is that they
48
49
50 18 might find a local optimum rather than a global optimum. Therefore, these algorithms should be
51
52 19 applied cautiously in stand management optimization. On the other hand, global optimization
53
54
55 20 algorithms may require prohibitively large numbers of functional evaluations (NFE). In other words,
56
57 21 more iterations are needed.

1
2
3
4 1
5
6
7 2 The dimensionality of decision variables and the convexity of objective function are the key factors
8
9
10 3 in stand management optimization (Roise 1986, Cao 2010). Pukkala (2009) recently proposed
11
12 4 population-based algorithms in stand management optimization, i.e., differential evolution (Storn and
13
14
15 5 Price 1997), particle swarm optimization (Kennedy and Eberhart 1995), evolution strategy (Bayer
16
17 6 and Schwefel 2002), Nelder-Mead (Nelder and Mead 1965). Population-based algorithms use
18
19
20 7 iteration technology beginning with a population of initial solution (referred to as individuals)
21
22
23 8 randomly generated, the whole population (or a part of it) is replaced by newly generated the best
24
25
26 9 individuals. The advantage of adopting population-based algorithms is the simplicity of convergence
27
28 10 criteria. For example, initial guesses, differentiability and smoothness of objective function are
29
30
31 11 unnecessary to be taken into consideration. These population-based algorithms have been
32
33 12 successfully applied to numerical optimization problems in many science and engineering disciplines
34
35
36 13 (Coello 2002), and have been further tested to solve stand management problems (Pukkala et al.
37
38 14 2010; Arias-Rodil et al. 2015) with empirical growth models. However, the previous studies either
39
40
41 15 simplified optimization problems, or applied relatively simple stand simulators. With more detailed
42
43
44 16 process-based models (thousands of state variables), and more complicated optimization problems (a
45
46 17 number of decision variables), the capability of population-based algorithms to optimize thinning
47
48
49 18 regimes remained unclear.
50
51
52 19
53
54
55 20 This study compared population-based algorithms linked with the process-based growth model
56
57
58 21 PipeQual (Mäkelä 1997; Mäkelä and Mäkinen 2003) in stand management optimization. The
59
60

1
2
3
4 1 objectives of this study were: 1) to evaluate the population-based algorithms for optimizing thinning
5
6 2 and rotation based on the process-based model; 2) to analyze effects of the number of thinning
7
8
9 3 decision variables on the performance of population-based algorithms.
10
11
12 4
13
14
15 5 **Materials and methods**
16
17
18 6 **Materials**
19
20
21
22 7 The biological data of seven simulated Scots pine (*Pinus sylvestris* L.) stands in Finnish conditions
23
24
25 8 (e.g., site types and temperature sum) were selected in this study (Table 1). These stands were
26
27 9 applied earlier in background calculations made for silvicultural recommendations in Finland
28
29
30 10 (Hyytiäinen et al. 2006). The initial age of stands varies from 20-29 years. The site type of the stands
31
32
33 11 covers *Myrtillus* (MT, stands 2-3), *Vaccinium* (VT, stands 4-7), and *Calluna* (CT, stand 1) sites. The
34
35 12 initial stand states present typical young Scots pine stands in Northern (stands 2, 4) and Southern
36
37
38 13 (stands 1, 3-6) Finland (Cao et al. 2015). The cost of logging was calculated by a logging model
39
40
41 14 (Kuitto et al. 1994), that involves more variables, such as productivity of felling, and on-site
42
43 15 transports, in addition to logging volume. This improves the accuracy of logging cost calculations.
44
45
46 16 The logging cost model consists of felling and transportation cost, as well as a fixed cost. The
47
48 17 average distance of traveling was 200 m. The default felling, transportation, and fixed costs, were
49
50
51 18 75.67 €/h, 53.35 €/h, and 100.00 €/h, respectively. The discount rate, roadside prices, and costs
52
53
54 19 were expressed in real terms. A 3% discount rate was constantly used. The roadside prices for
55
56 20 sawlog was 52.98 €/m³, and pulpwood 26.24 €/m³ (Hyytiäinen et al. 2004). The unit silviculture
57
58
59 21 cost for soil preparation was 142 €/ha, sowing 600 €/ha, and other silviculture operation (tending
60

and slashing) was 276 €/ha (Cao et al. 2010). Soil preparation was carried out in the first year for all stands, and sowing in the following year except stand 1 which was naturally regenerated. According to Finnish silvicultural recommendations, depending on temperature sum, site types and regeneration methods, we assumed that the selected stands were tended at ages 20, 15, 18, 13, and 16 for stands 1, 2, 3, 4, and stands 5-7, respectively (Hyytiäinen et al. 2006). A more detailed description of silviculture cost in Finnish conditions was presented in Cao et al. (2010).

The process-based model

The process-based growth model PipeQual (Mäkelä 1997, 2002; Mäkelä and Mäkinen 2003) has been integrated into the OptiFor simulation-optimization system in which Osyczka's direct and random search (DRS) is the optimizer, and the PipeQual model is the simulator (Cao 2010). The advantage of using PipeQual is that the inputs of PipeQual are common initial stand states, while most of other process models require more climatic and soil inputs. PipeQual is a dynamic growth and wood quality model that derives tree growth from carbon acquisition and allocation in a process-based framework. It also contains a detailed semi-empirical description of the development of stem structure and branchiness that allows for the model to be applied to predictions of wood quality in individual stems as influenced by forest management. The model is constructed in a modular manner (Mäkelä 2003), with separate modules for the whole tree (CROBAS, Mäkelä 1997), vertical structure (WHORL) and branches in whorls (BRANCH). The stand is composed of a number of size classes (here 10), each of which is simulated by its mean tree in a distance-independent setting. The description of tree structure in PipeQual largely derives from the pipe model (Shinozaki et al. 1964a,

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

b), profile theory (Chiba et al. 1988) and fractal crown allometry (Mäkelä and Sievänen 1992; Duursma et al. 2010). At the tree level, state variables include the biomasses of foliage, fine roots, stem, branches, and transport roots, as well as stem and crown dimensional variables. WHORL includes the height, stem and total branch cross-sectional area and foliage mass of each whorl, while BRANCH decomposes the branch area into individual branches. Growth is calculated from photosynthesis and respiration at the tree level. Tree annual growth is used as the input of the whorl and branch levels to describe the growth and senescence of sapwood and branches. The model structure of PipeQual was illustrated in Mäkelä and Mäkinen (2003, Fig. 1). The model has been parameterized for *Pinus sylvestris* and *Picea abies* (Kantola et al. 2007; Mäkelä et al. 2016), and it has been applied to economic optimization in both species (Hyytiäinen et al. 2004; Cao et al. 2010, 2015; Hurttala et al. 2017).

The optimization problem

We formulated a bound-constrained optimization problem (eqs. 1-3) for optimizing thinning regimes of forest stands. The negative bare land value (-BLV) of a stand as the objective function $f(t,H|Z(t_0))$ was minimized (i.e. maximization of BLV) by changing decision variables t (time of the j th thinning, yr) and H (proportion of trees harvested in tree size class m at the j th thinning), in the condition of stand states Z (state variables) at initial age t_0 .

$$\min f(t,H|Z(t_0)) \tag{1}$$

$$s.t. \quad \mathbf{t}=(t_1,t_2,\dots,t_{n+1}) \quad t_j \in [1,25] \quad (2)$$

$$\mathbf{H}=(h_{ji})_{n \times 3} \quad h_{ji} \in [0,1] \quad (3)$$

where h_{ji} is thinning rate defined by linearly interpolating in the i th tree size class at the j th thinning.

The BLV of a thinning regime for timber production can be written as follows (eq. 4):

$$BLV = \frac{\sum_{j=1}^{n+1} [\sum_{l=1}^{10} \sum_{w=1}^2 p_w v_{jtw} - c_j] e^{-rt_j} - c_0}{1 - e^{-rt_{n+1}}} \quad (4)$$

where p_w denotes the timber prices for timber categories ($w=1$ pulpwood, $w=2$ sawlog), v denotes harvested timber volume, c_j is the logging cost at the j th harvest, $j=n+1$ means final harvest, c_0 is the discounted stand establishment cost, and r denotes discount rate.

Optimization algorithms

The criteria of algorithm evaluation applied in this study were the accuracy, efficiency and robustness of optimization algorithms. The accuracy of algorithm is how close one objective function value to the standard value that was defined as the objective function value of direct and random search in this study. The efficiency of algorithm is the performance of an algorithm based on the number of computation resources that can be evaluated by using time complexity (time consumption by running an algorithm). Robustness is another kind of performance of an algorithm evaluated by

the capacity of tolerating errors of inputs.

The optimization algorithms tested in this study were Particle Swarm Optimization (PSO), Differential Evolution (DE), Evolution Strategy (ES), Nelder-Mead (NM), and Direct and Random Search (DRS). The optimization algorithms were programmed as candidate solutions of optimal thinning regimes. The candidate solutions were evaluated with the objective function by calling the process-based growth model. The DRS algorithm starts with a vector of initial points (a candidate solution \mathbf{x}_i). The population-based algorithms (PSO, DE, ES, NM) begin with an initial population of m individuals (initial solutions), each individual is an n -dimensional vector, in which x_{ij} is the j th decision variable of the i th candidate solution \mathbf{x}_i . The initial solution $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})$ were randomly generated from the feasible region $[L_j, U_j]$ of decision variables, where L_j is the lower bound, and U_j the upper bound of the j th decision variable. The convergence criterion of the DRS algorithm was the minimum difference ε (difference of two candidate solutions \mathbf{x}_{i+1} and \mathbf{x}_i) in search step size (if $\|\mathbf{x}_{i+1} - \mathbf{x}_i\| < \varepsilon$, stop). In contrast, the convergence criteria of population-based algorithms were either the minimum difference ε in objective function values (if $\|f(\mathbf{x}_{i+1}) - f(\mathbf{x}_i)\| < \varepsilon$, stop), or a maximum iteration number achieved.

In this study, the number of individuals of population (n_{pop}) and maximum iteration number (n_{it}) were modified from Pukkala (2009) due to convergence problems raised by using the more complicated process-based growth model. Therefore, the values of n_{it} were increased to 8000 and 1500 for ES and NM, respectively. The value of n_{pop} was $8 \times$ number of decision variables (n_d). We

repeated 50 times initiating different candidate solutions to find optimal solutions. Decision variables include timing, frequency, intensity, type of thinning, and the length of rotation. The number of decision variables were defined by the number of thinning (n_{thin}) and the type of thinning (n_{type} , the number of thinning points defining thinning intensity by tree size classes), that is $n_d = n_{thin}(n_{type}+1)+1$.

Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic global optimization algorithm inspired by swarm behavior in birds, insects, fish, even human behavior (Kennedy and Eberhart 1995). In PSO, each particle (individual) adjusts its position and velocity, moves to some global objective through information exchange between its neighbor particles and the whole swarm (population). PSO carries out a five-step search:

1) Randomly generate initial swarm (population) which consists of $n_{pop}=10 \times n_d + 50$ particles (individuals), each particle $\mathbf{x}_i=(x_{i1}, x_{i2}, \dots, x_{in})$ has velocity $\mathbf{v}_i=(v_{i1}, v_{i2}, \dots, v_{in})$. 2) Evaluate each particle, store the previous best position for each particle $\mathbf{pbest}_i=(p_{i1}, p_{i2}, \dots, p_{in})$, and find the global best for the entire population $\mathbf{gbest}=(g_1, g_2, \dots, g_n)$. 3) Update the $i+1$ th generation $\mathbf{x}_{i+1}=\mathbf{x}_i+\mathbf{v}_{i+1}$, where \mathbf{v}_{i+1} is updated as

$$\mathbf{v}_{i+1} = w\mathbf{v}_i + c_1 r_1 (\mathbf{pbest}_i - \mathbf{x}_i) + c_2 r_2 (\mathbf{gbest} - \mathbf{x}_i), \quad (5)$$

where $w=(0.4+(0.9-0.4)(n_{it}-i)/n_{it})$ is the inertia factor decreased linearly from 0.9 to 0.4, n_{it} is number of iterations, $c_1=1.5$ and $c_2=1.5$ are constants called cognitive and social parameters, respectively, r_1 , r_2 are random values between $[0,1]$ (eq. 5). 4) Evaluate every new particle, $\mathbf{x}_i = \mathbf{x}_{i+1}$ if $f(\mathbf{x}_{i+1}) < f(\mathbf{x}_i)$, otherwise $\mathbf{x}_{i+1} = \mathbf{x}_i$. Compare the best value $f(\mathbf{x}_{i+1})$ with $f(\mathbf{pbest}_i)$ and $f(\mathbf{gbest})$, if $f(\mathbf{x}_{i+1}) < f(\mathbf{pbest}_i)$, $\mathbf{pbest}_{i+1} = \mathbf{x}_{i+1}$, if $f(\mathbf{x}_{i+1}) < f(\mathbf{gbest})$, $\mathbf{gbest} = \mathbf{x}_{i+1}$. 5) Check whether the number of iterations ($n_{it}=50$) reaches up its maximum limit. If not, go to step 3.

Differential Evolution

Storn and Price (1997) proposed Differential Evolution (DE), a stochastic evolutionary algorithm to solve global optimization problems. In DE an offspring individual (candidate solution) is generated through mutation and crossover with the weighted difference of parent solutions. The offspring may replace its parent through competitive selection. The most applied mutation strategies are rand/1, best/1, current to best/1, best/2, and rand/2 schemes (for details, see Liu et al. 2010). In this version we used the mutation strategy of current to best/1 scheme rather than the rand/1 scheme used in Pukkala (2009). The method of Differential Evolution (DE) performs a six-step search:

- 1) Randomly generate initial population whose value is $n_{pop} = 5 \times n_d$ (initial parent individuals) $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})$.
- 2) Evaluate the initial population, calculate every individual function value $f(\mathbf{x}_i)$, and record the optimized value and the previous best individual \mathbf{pbest}_i .
- 3) Randomly select two remainder individuals \mathbf{x}_{r1} and \mathbf{x}_{r2} , and calculate for the mutant individual $\mathbf{y}_i = (y_{i1}, y_{i2}, \dots, y_{in})$ using the current to best/1 scheme (eq. 6),

$$\mathbf{y}_i = \mathbf{x}_i + \alpha(\mathbf{pbest}_i - \mathbf{x}_i) + \beta(\mathbf{x}_{r1} - \mathbf{x}_{r2}), \quad (6)$$

where α is a random number between $[0,1]$, $\beta=0.8$. 4) Generate the offspring individual \mathbf{x}_i' by a crossover operation on \mathbf{x}_i and \mathbf{y}_i with a crossover probability parameter CR (in this study $CR=0.5$) determining the genes of \mathbf{x}_i' are inherited from \mathbf{x}_i or \mathbf{y}_i . Let $x_{ij}' = y_{ij}$, if a random real number from $[0,1]$ is smaller than CR , otherwise, $x_{ij}' = x_{ij}$. 5) Select the best individual for the next generation \mathbf{x}_{i+1} by the competition between the offspring individual \mathbf{x}_i' and the parent individual \mathbf{x}_i . If $f(\mathbf{x}_i') \leq f(\mathbf{x}_i)$, $\mathbf{x}_{i+1} = \mathbf{x}_i'$, otherwise, $\mathbf{x}_{i+1} = \mathbf{x}_i$. 6) Check whether the number of iterations ($n_{it}=100$) reaches the maximum limit. If not, go to step 3.

Evolution Strategy (ES)

The Evolution Strategy (ES) uses strategy parameters to determine how a recombinant is mutated. ES generates an offspring as a mutated recombination from two parents. One of the parents is the previous best individual, and the other one is randomly drawn from the remaining individuals. The offspring then compete with the parents. If the offspring is better, the mutated solution replaces the worst solution of the parent population. The best solution at the last generation is the optimal solution. ES conducts a five-step search:

- 1) Randomly generate initial population \mathbf{x}_i whose value n_{pop} equal to $10 \times n_d + 50$. The initial strategy parameters σ_i was calculated from $\sigma_i = \alpha \mathbf{x}_i$, where $\alpha=0.2$. 2) Obtain the previous best individual \mathbf{pbest}_i and σ_{best_i} strategy parameter values. Recombine the selected parents, the best individual \mathbf{pbest}_i and random individual \mathbf{x}_r , obtain the recombined individual $\mathbf{x}_m = 0.5(\mathbf{pbest}_i + \mathbf{x}_r)$, and strategy parameters

$$\sigma_m = 0.5(\sigma_{best_i} + \sigma_r) \times \exp(\tau_g \times N(0,1) + \tau_l \times N(0,1)), \quad (7)$$

where the global study parameter τ_g is $1/\sqrt{(2 \times n_d)}$, the local study parameter τ_l is $1/\sqrt{(2 \times n_d)}$, $N(0,1)$ is a normally distributed random number. 3) Mutate an offspring individual $\mathbf{x}' = \mathbf{x}_m + \sigma_m \times N(0,1)$. 4) Evaluate the new individual \mathbf{x}' . Replace the worst solution of the parent generation if $f(\mathbf{x}')$ is less than the worst function value. 5) Check whether the number of iterations ($n_{it}=8000$) reaches its maximum limit. If not, go to step 2.

Nelder-Mead

Similar to ES, Nelder-Mead (NM) also uses a new candidate solution to replace the worst solution of all solutions at every iteration. In NM the new candidate solution is calculated based on the centroid solution and the best solution through reflection, expansion, and contraction operations. In case none of better new candidate solutions can be found in the reflection, expansion, and contraction operations, NM carries out an additional shrinking operation for a new iteration by updating all candidate solutions except the best solution. In NM all operations are calculated without stochasticity. NM implements a six-step search (Lagarias et al. 1998):

- 1) Randomly generate initial population whose value n_{pop} equals to $8 \times n_d$. Select the best, the worst, the second worst solutions \mathbf{x}_b , \mathbf{x}_w , \mathbf{x}_{sw} from all candidate solutions by their function values $f(\mathbf{x}_b)$, $f(\mathbf{x}_w)$, $f(\mathbf{x}_{sw})$. 2) Calculate the reflection point $\mathbf{x}_{rf} = (1 + \rho)\mathbf{x}_m - \rho\mathbf{x}_w$, where the reflection parameter $\rho = 1.4$ and the centroid point (average except the worst point \mathbf{x}_w) $\mathbf{x}_m = \sum_{i \neq w} (\mathbf{x}_i / (n_d - 1))$. If $f(\mathbf{x}_b) < f(\mathbf{x}_{rf}) < f(\mathbf{x}_{sw})$,

1 replace \mathbf{x}_w , with \mathbf{x}_{rf} and terminate the iteration. 3) If $f(\mathbf{x}_{rf}) < f(\mathbf{x}_b)$, compute expansion point $\mathbf{x}_e = \chi \mathbf{x}_{rf} + (1 -$
 2 $\chi) \mathbf{x}_m$, where the expansion parameter $\chi = 2.5$. If $f(\mathbf{x}_e) \leq f(\mathbf{x}_b)$, replace \mathbf{x}_w with \mathbf{x}_e and terminate the
 3 iteration; else replace \mathbf{x}_w , with \mathbf{x}_{rf} and terminate the iteration. 4) If $f(\mathbf{x}_{rf}) > f(\mathbf{x}_{sw})$, compute inside
 4 contraction point $\mathbf{x}_c = \gamma \mathbf{x}_w + (1 - \gamma) \mathbf{x}_m$, where the contract parameter $\gamma = 0.5$. If $f(\mathbf{x}_c) \leq f(\mathbf{x}_w)$, replace \mathbf{x}_w with
 5 \mathbf{x}_c and terminate the iteration, else go to step 5. If $f(\mathbf{x}_{sw}) \leq f(\mathbf{x}_{rf}) < f(\mathbf{x}_w)$, compute outside contraction
 6 point $\mathbf{x}_c = \gamma \mathbf{x}_{rf} + (1 - \gamma) \mathbf{x}_m$. If $f(\mathbf{x}_c) \leq f(\mathbf{x}_{rf})$, replace \mathbf{x}_w with \mathbf{x}_c and terminate the iteration, else go to step 5.
 7 5) Compute $\mathbf{x}_i (i \neq b)$ with \mathbf{x}_b and shrinkage parameter $\delta = 0.8$ for the new generation $\mathbf{x}_i' = \mathbf{x}_b + \delta(\mathbf{x}_i - \mathbf{x}_b)$,
 8 and begin a new iteration. 6) Check whether the number of iterations ($n_{it} = 1500$) reaches its
 9 maximum limit. If not, go to step 2.

11 *Direct and Random Search*

12 In this study, Osyczka's (1984) direct and random search (DRS) was used as a reference algorithm
 13 that is a modified version of Hooke and Jeeves' method (1961). The Hooke and Jeeves' method has
 14 been earlier applied in stand management optimization (e.g. Roise 1986, Haight and Monserud
 15 1990). Osyczka (1984) modified the version of discrete steps of the Hooke and Jeeves' method
 16 (1961) to overcome local optimum problems. The Osyczka's (1984) direct and random search
 17 algorithm (DRS) is a hybrid algorithm based on neighborhood search, shotgun search and Hooke and
 18 Jeeves' direct search. By integrating neighborhood search and random search into direct search
 19 phases, DRS has been proved to be a successful method for solving forest management problems
 20 (e.g. Hyytiäinen et al. 2005, Cao et al. 2010, Hurttala et al. 2017).

22 In this study we initiated initial points 30 times and then calculated for each optimal solution. We

analyzed the speed of convergence, the time complexity of algorithms, the rate of successful search, and the sensitivity of decision variables for population-based algorithms. The capital O notation expresses the time complexity of an algorithm excluding coefficients and lower order terms, n_{it} is maximum iteration numbers, n_{pop} is number of individuals of population (generation), and n_d is number of decision variables. The time complexity $T(n)$ of ES was $T(n) = O(n_{it} \times n_d)$, the time complexity of DE and PSO was $T(n) = O(n_{it} \times n_{pop} \times n_d)$. The worse-case (defined as the maximum amount of spent time) time complexity of NM was $T(n) = O(n_{it} \times n_{pop} \times n_d)$, and the best-case time complexity of NM was $T(n) = O(n_{it} \times n_d)$. The rate of successful search was defined using $(n_{suc}/n_{run}) \times 100\%$, where n_{run} is the number of runs, n_{suc} is the number of successful search. A successful search is achieved when the relative errors of optimal values between population-based algorithms and the reference algorithm DRS is less than 0.01. In this study we conducted sensitivity analysis by increasing the number of decision variables to analyze the effects of the number of decision variables on the objective function value, the number of functional evaluations, and the amount of central processing unit time. The sensitivity analysis of decision variables was therefore designed based on the equation $n_d = n_{thin}(n_{type}+1)+1$, as $2(2+1)+1=7$, $3(2+1)+1=10$, $3(3+1)+1=13$, $5(2+1)+1=16$, $6(2+1)+1=19$, $5(3+1)+1=21$, and $6(3+1)+1=25$ variables.

Results

Optimal solutions

The results showed that both DE and PSO can successfully discover the highest objective function values from our data. For instance, DE was superior in stands 1, 4, and 7, and PSO was superior in

stands 1-3, and 7 to the competing algorithms. ES and DRS also obtained the highest objective function values in stands 6, and 5, respectively. Nevertheless, NM found none of the highest objective function values. DE and DRS never resulted in the lowest objective function values. However, PSO (stand 4), ES (stand 2), and NM (stands 1, 3, 5-7) led to the lowest objective function values (Table 2).

Our results showed that the variation of optimal rotation was 1-15 years. The shortest rotations for selected stands were obtained by PSO (stands 2, 5, 7), ES (stands 4, 6), DE (stands 2, 5), NM (stands 1-3), and DRS (stand 5). PSO and DE were equally good at searching the highest objective function value for stand 7 (2110 €/ha). However, DE resulted in a little longer rotation (93 yrs). Although all the objective function values by NM were lower than the other algorithms, NM resulted in shorter rotations except stand 4 (Table 2).

The results showed that basal area development increased at the beginning, and then decreased after one or two early thinnings for the Scots Pine stands examined. The highest basal area was 22.5-28.0 m²/ha, and the lowest basal area before clearcut was 7.2-16.7 m²/ha depending on the initial stand states (Fig. 1). The more fertile the site (H100 index) was, the earlier the first thinning (Fig. 1a), and the denser the stand was, the earlier the first thinning (Fig. 1b). The timing of the first thinnings varied from 33 to 54 yrs, while the basal area before the first thinnings was in the range of 23.4-27.3 m²/ha (Fig. 1).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

The results revealed that thinning frequency was quite consistent. The optimal number of thinnings was three or four for all tested stands (Table 2). For sparse stands with initial density 1500 trees/ha, the optimal number of thinnings was three for stands 1, 2, 5, and four for stands 3-4. For denser stands with initial density 2000 trees/ha (stand 6) and 3000 trees/ha (stand 7), however, the optimal number of thinnings was always four.

According to our results, the type of thinning significantly changed in selecting different tree size classes to be removed from early precommercial thinnings to later rotation thinnings. For early thinnings most small and medium size trees were remained, and only some large size trees were thinned. For later thinnings large and medium size trees were mostly removed, and some of small size trees were selectively thinned. However, the optimal thinning type varied depending on the algorithms. Figure 2 shows an exception of optimal solutions for stand 6 by thinning type. For instance, ES resulted in thinning from medium size classes at the 4th thinning (Fig. 2d).

Accuracy of algorithms

With maximum 17 decision variables ($n_{thin}=4$, $n_{type}=3$, $n_d=17$), the optimal solutions of all the algorithms were satisfactory in accuracy for sparse stands with initial density 1500 trees/ha. However, the differences became somewhat serious for denser stands with initial density 2000-3000 trees/ha (Table 2). The differences of the optimized objective function values found by all the

algorithms were less than 1% (0.17-0.82%) for stands 1-5 with initial density 1500 trees/ha. For stand 6 with initial density 2000 trees/ha, and stand 7 with initial density 3000 trees/ha, the errors were enlarged to 1.86%, and 2.13%, respectively (Table 3). From the perspective of accuracy, PSO was the most accurate algorithm that resulted in only 0.00-0.82% differences of optimized objective function values for all stands examined. For sparse stands (stands 1-5), the most accurate one was DRS which led to 0.00-0.29% errors only (Table 3).

Efficiency of algorithms

Because of the high level of detail in forest stand projection in our study, the vast majority of computation time was spent in computing stand projections (0.14 seconds were spent on calculating the objective function value for one time using Compaq Visual Fortran (version 6.6), an Intel (R) Core (TM) i5-3470 processor at 3.2Ghz and 4.00 GB of RAM memory). Therefore, the number of functional evaluations is a direct measure of overall computing time. The speed of convergence measured by the number of functional evaluations indicated that NM (Fig. 3, solid line) was the fastest algorithm to converge, followed by PSO and ES, while DE was the slowest one (Fig. 3).

On average, the number of functional evaluations for NM was 4500, DE 6500 and ES 8000. The number of functional evaluations for PSO was 9000, which is about twice that of NM (Fig. 3). The most efficient algorithm was NM, which required about 600 seconds with the best-time complexity $O(n_{it} \times n_d)$. DE and ES both required about 250 and 500 seconds more than NM, respectively. From

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

the perspective of convergence, four algorithms were all efficient (Fig. 4). DE ($n_{pop}=65$) converged to the optimum with 70 iterations (Fig. 4a), PSO ($n_{pop}=180$) converged with 25 iterations (Fig. 4b), ES ($n_{pop}=180$) converged with 3000 iterations (Fig. 4c), and NM ($n_{pop}=104$) converged with 800 iterations (Fig. 4d).

Robustness of algorithms

Our results showed that DE was clearly dominant in terms of robustness (Fig.5). The rate of successful search of DE in 50 runs was 100%, followed by NM 90% and PSO 90% but, ES achieved 82% successful search only. The optimal solutions of population-based algorithm varied with different random numbers in 50 runs of optimization calculations. The optimal solutions of DE and DRS were clearly converged with small variations (Fig. 5). In contrast, ES resulted in larger variations. In other words, PSO, NM and ES led to local optima in 50 runs. Nevertheless, the difference between minimum and mean values for ES and NM were smaller than that of DE and PSO (Fig. 4c, d).

According to our results, variations enlarged with increasing the number of decision variables (Table 4, Fig. 6). With five decision variables ($n_{thin}=1, n_{type}=3, n_d=5$) the difference in relative objective function values was insignificant (0.0%). Increasing decision variables to 25 ($n_{thin}=6, n_{type}=3, n_d=25$) led to changes in the relative objective function values that were significantly greater (2.7%) than that of five decision variables (Table 4). Among five algorithms, DE, PSO, ES and DRS generated

higher objective function values with increasing number of decision variables. Especially, PSO was able to find the highest objective value even with 25 decision variables (Fig. 6). Nevertheless, it seems that NM suffered difficulties in escaping local optima when the number of decision variables increased above 16 ($n_{thin}=5$, $n_{type}=2$, $n_d=16$). For example, NM only found three thinnings for stand 3, but the other four algorithms (PSO, DE, ES, and DRS) obtained four thinnings to be optimal.

Discussion

In general, PSO was clearly dominant in searching ability compared to the other algorithms in this study. Including DRS, all the algorithms we tested were fairly successful in searching optimal solutions. The differences in optimal solutions were caused by the search capability of algorithms in terms of decision variables, such as the timing, frequency, and type of thinning, as well as the length of rotation. For example, NM only found three thinnings to be optimal, and this led to a shorter rotation length for stand 3. Both ES and DRS found four thinnings to be optimal. However, ES led to earlier thinnings and a shorter rotation (83 yrs) than those of DRS (92 yrs) for stand 6. Meanwhile, the type of thinning by ES tended to leave more trees in sawlog tree classes for later harvesting (Fig. 2c, d). This was especially true at the fourth thinning: the type of thinning was thinning from middle tree size classes by ES (Fig. 2d). This implies that more sawlog trees were expected at final harvest. It should be noted that, a 3% discount rate was constantly used in this study. A higher discount rate may lead to shorter rotations, and vice versa.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1 It is hard to say that one optimization algorithm is clearly superior over other algorithms in any
2
3
4
5
6
7 2 situations for forest management problems (Roise 1986, Pukkala 2009), or for applications in other
8
9
10 3 fields (Lv et al. 2015). DE, PSO and ES obtained the highest objective values in some stands, while
11
12 4 NM never found the highest objective values. However, there were only slight differences among
13
14 5 DE, PSO and ES in terms of accuracy (Table 2). Sofge et al. (2002) compared seven evolutionary
15
16
17 6 algorithms for a two-level optimization problem. They also found the differences in accuracy
18
19
20 7 between evolutionary algorithms were quite small. This is in line with Pukkala (2009) for stand
21
22 8 management optimization that differences between HJ and the population-based algorithms were
23
24
25 9 rather small.
26
27
28 10
29
30
31 11 The conventional NM was still the fastest algorithm in this study. By contrast, DE and PSO were
32
33
34 12 somewhat disappointing in terms of convergence speed (DE) and time consumption (PSO). Our
35
36
37 13 results confirmed Fan and Zahara (2007) that the convergence of NM was faster than that of PSO
38
39 14 based on Powell badly scaled function. The results were different from Sofge et al. (2002) that the
40
41
42 15 convergence of ES was faster than that of PSO. Nevertheless, the parameters and operation of PSO
43
44 16 and ES used in their multiple travelling salesman problem were different from this study. This was in
45
46
47 17 line with Vesterstrøm and Thomsen (2004), especially for those optimization problems where the
48
49
50 18 number of dimensions for the search space is relatively low.
51
52
53 19
54
55
56 20 In this study, the time consumption of NM was less than that of PSO, which was in line with Pukkala
57
58
59 21 (2009) and Arial-Rodil (2015). The time consumption of PSO on average was about two times more

than that of NM, because the number of function evaluations (NFE) of PSO was much more than that of NM. In PSO the maximum number of iterations was set as 50 with the number of population individuals 180. All individuals were evaluated at each iteration, therefore, the optimizer called the stand simulator about 9000 (50×180) times. In NM the initialization step selected the best, the worst, and the second worst individuals by evaluating all initial population individuals (56-136). Then only the previous worst one should be replaced by evaluating 1-4 individuals (reflection, expansion, inside contraction or outside contraction points) at iterations. The maximum number of iterations was set as 1500 in NM. Therefore, the optimizer would call the stand simulator 1556-6136 times. This explains the performance of algorithms in terms of time consumption.

Our results demonstrated that DE was clearly the most robust algorithm (Fig. 5). This is in agreement with Arial-Rodil (2015) and Vesterstrøm and Thomsen (2004) who found that DE was more robust in comparison to PSO and other evolutionary algorithms, because PSO was more dependent on the random numbers than DE. The number of dimensions for the search space in this study, i.e., the number of decision variables, may also affect the robustness of algorithms. As illustrated in Figure 6, NM was rather sensitive to the number of decision variables. This implies that PSO, DE and ES might be more suitable for such optimization problems where the number of dimensions for the search space is relatively high (Table 4).

Various versions of the original population-based algorithms have been developed or modified for numerical optimization studies. In this study we applied different operation methods for DE, PSO,

1
2
3
4 1 ES and NM from Pukkala (2009) to improve the optimal solutions. The way to generate initial
5
6 2 populations in NM was the same as in Pukkala (2009). However, in NM we set different parameters
7
8
9 3 of reflection, expansion, and shrink operations rather than those suggested parameter values in
10
11
12 4 Nelder and Mead (1965), Lagarais et al. (1998), as well as Wang and Shoup (2011). As a result, for
13
14 5 example, the objective value increased 1% for stand 5 when the number of decision variables was 13.
15
16
17 6
18
19
20
21 7 Compared to Evolutionary Algorithms and PSO, DE has shown superior performance in several real-
22
23 8 world applications as well (Vesterström and Thomsen 2004). In the applications of stand
24
25
26 9 management optimization, DE was found superior for even-aged stand management problems
27
28
29 10 (Pukkala 2009; Airas-Radil et al. 2015), but PSO might be superior when diameter structures were
30
31 11 used as a penalty function for uneven-aged stand management (Pukkala et al. 2010). Therefore, the
32
33
34 12 complexity of optimization problems might affect the performance of optimization algorithms. The
35
36
37 13 accuracy, efficiency, and robustness of these algorithms vary, depending on the complexity of
38
39 14 optimization problems and stand simulators. Because the process-based model computes tree growth
40
41
42 15 as biomass accumulation in trees by photosynthesis and respiration depending on the development of
43
44 16 tree crowns and foliage mass, the process-based model is computationally demanding compared to
45
46
47 17 empirical growth models. The total variable number of a stand in the process-based model PipeQual
48
49
50 18 (Mäkelä and Mäkinen 2003) used in this study is 7920-18320 state variables (Hyytiäinen et al. 2004).
51
52 19 This is enormous compared with the individual-tree model with 37-65 state variables (Cao 2003)
53
54
55 20 tested by Pukkala (2009), and the whole-stand model applied in Airas-Radil et al. (2015) with three
56
57 21 state variables only. In fact, dynamic programming is efficient enough for whole-stand models when
58
59
60

the number of state variables is small. Therefore, the dimensionality of optimization problem based on a whole-stand model in Airas-Radil et al. (2015) might be too simple for the population-based algorithms.

In this study we only tested five site types with initial density 1500 trees/ha (stands 1-5), and one site type with initial density 1500, 2000, and 3000 trees/ha (stands 5-7). The denser the initial stand density is, the more frequent thinnings are needed (Cao et al. 2006). It would be interesting to study the effects of initial stand states by testing different site types with various stand densities. However, this is out of the scope of this study. In addition, a modified hybrid algorithm may significantly improve search efficiency and the quality of resulting solutions. For instance, Fan and Zahara (2007) suggested that the slow convergence of PSO could be improved by combining NM to the hybrid NM-PSO algorithm in unconstrained optimization problems. In constrained numerical and engineering optimization problems, Liu et al. (2010) also found that the traditional PSO easily fell into local optima whereas this could be improved by the hybrid algorithm PSO-DE they proposed as well. Recently, similar hybrid algorithms have also been proposed in ecological modelling studies, for example, applying back propagation-genetic algorithm to predict soil temperature (Kazemi 2018), and support vector machine-firefly algorithm to predict water balance (Moazenzadeh 2018). These hybrid algorithms might be useful to improve the convergence speed, and to overcome local optima in stand management optimization. Therefore, the application of hybrid algorithms for stand management optimization would be a promising study as well.

1
2
3
4 1 In conclusion, DE (Differential Evolution), PSO (Particle Swarm Optimization) and ES (Evolution
5
6 2 Strategy) were mostly superior to NM (Nelder-Mead) in stand management optimization when stand
7
8
9 3 development was simulated using a process-based growth model. Among these tested algorithms,
10
11
12 4 PSO was the most accurate algorithm, DE was the most robust, and NM was the most efficient. DE
13
14 5 would be an effective algorithm as an alternative in stand management optimization.
15
16
17
18 6
19

20 7 **References**

21
22
23 8 Arias-Rodil M, Pukkala T, González-González JM, Barrio-Anta M, Diéguez-Aranda U. 2015. Use of
24
25 9 depth-first search and direct search methods to optimize even-aged stand management: a
26
27
28 10 case study involving maritime pine in Asturias (northwest Spain). Can J For Res. 45(10):
29
30
31 11 1269-1279.
32
33
34 12 Bayer H-G, Schwefel H-P. 2002. Evolution strategies: A comprehensive introduction. Nat Comput.
35
36 13 1(1): 3-52.
37
38
39
40 14 Brodie JD, Haight RG. 1985. Optimization of silvicultural investment for several types of stand
41
42
43 15 projection systems. Can J For Res. 15(1): 188-191.
44
45
46 16 Cao T. 2003. Optimal harvesting for even-aged Norway spruce stands using an individual-tree
47
48 17 growth model [Research paper 897]. Vantaa: The Finnish Forest Research Institute.
49
50
51
52 18 Cao T, Hyytiäinen K, Tahvonen O, Valsta L. 2006. Effects of initial stand states on optimal thinning
53
54 19 regime and rotation of *Picea abies* stands. Scand J For Res. 21(5): 388-398.
55
56
57
58 20 Cao T. 2010. Silvicultural decisions based on simulation-optimization systems [Dissertation].
59
60

Helsinki: Finnish Society of Forest Science.

- Cao T, Valsta L, Mäkelä A. 2010. A comparison of carbon assessment methods for optimizing timber production and carbon sequestration in Scots pine stands. *For Ecol Manage.* 260(10): 1726-1734.
- Cao T, Hyytiäinen K, Hurttala H, Valsta L, Vanclay JK. 2015. An integrated assessment approach to optimal forest bioenergy production for young Scots pine stands. *Forest Ecosyst.* 2: 1-19.
- Chiba Y, Fujimori T, Kiyono Y. 1988. Another interpretation of the profile diagram and its availability with consideration of the growth process of forest trees. *J Jpn For Soc.* 70: 245-254.
- Coello CAC. 2002. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Comput Methods Appl Mech Engrg.* 191(11): 1245-1287.
- Duursma RA, Mäkelä A, Reid DEB, Jokela EJ, Porté A, Roberts SD. 2010. Self-shading affects allometric scaling in trees. *Functional Ecology.* 24: 723-730.
- Fan SKS, Zahara E. 2007. A hybrid simplex search and particle swarm optimization for unconstrained optimization. *Euro J Oper Res.* 181(2): 527-548.
- Haight RG, Monserud RA. 1990. Optimizing any-aged management of mixed-species stands: II. Effects of decision criteria. *For Sci.* 36(1): 125-144.
- Hann DW, Brodie JD. 1980. Even-aged management: basic managerial questions and available or potential techniques for answering them. General Technical Report, Intermountain Forest

- and Range Experiment Station, USDA Forest Service. INT-83.
- Hooke R, Jeeves TA. 1961. "Direct search" solution of numerical and statistical problems. J Assoc Comput Mach. 8(2): 212-229.
- Hurtala H, Cao T, Valsta L. 2017. Optimization of Scots pine (*Pinus sylvestris*) management with the total net return from the value chain. J For Econ. 28: 1-11.
- Hyytiäinen K, Hari P, Kokkila T, Mäkelä A, Tahvonen O, Taipale J. 2004. Connecting a process-based forest growth model to stand-level economic optimization. Can J For Res. 34(10): 2060-2073.
- Hyytiäinen K, Ilomäki S, Mäkelä A, Kinnunen K. 2006. Economic analysis of stand establishment for Scots pine. Can J For Res. 36(5): 1179-1189.
- Kantola A, Mäkinen H, Mäkelä A. 2007. Stem form and branchiness of Norway spruce as sawn timber – predicted by a process-based model. For Ecol Manage. 241: 209-222.
- Kazemi, SMR., Biodgoli BM, Shamshirband S, Karimi SM, Ghorbani MA, Chau K. Pour RK. 2018. Novel genetic-based negative correlation learning for estimating soil temperature. Eng Appl Comp Fluid. 12 (1):506-516.
- Kennedy J, Eberhart RC. 1995. Particle swarm optimization. Proceedings of the IEEE international conference on neural networks; Nov 27-Dec 1; Perth, Australia. p. 1942–1948.
- Kuitto P, Keskinen S, Lindroos J, Oijala T, Rajamäki J, Räsänen T, Teräväinen J. 1994. Mechanized cutting and forest haulage. Painovalmiste KY, Helsinki. Metsäteho Report 410. (In Finnish).

- 1
2
3
4 1 Lagarias JC, Reeds JA, Wright MH, Wright PE. 1998. Convergence properties of the Nelder–Mead
5
6 2 simplex method in low dimensions. *SIAM J Optim.* 9(1): 112–147.
7
8
9
10 3 Landsberg JJ, Waring RH. 1997. A generalized model of forest productivity using simplified
11
12 4 concepts of radiation-use efficiency, carbon balance and partitioning. *For Ecol Manage.*
13
14 5 95(3): 209-228.
15
16
17
18 6 Lindner M, Sievänen R, Pretzsch H. 1997. Improving the simulation of stand structure in a forest gap
19
20 7 model. *For Ecol Manage.* 95: 183-195.
21
22
23
24 8 Liu H, Cai Z, Wang Y. 2010. Hybridizing particle swarm optimization with differential evolution for
25
26 9 constrained numerical and engineering optimization. *Appl Soft Comput.* 10(2): 629-640.
27
28
29
30 10 Lockwood C, Moore T. 1993. Harvest scheduling with spatial constraints: a simulated annealing
31
32 11 approach. *Can J For Res.* 23(3): 468-478.
33
34
35
36 12 Lu F, Eriksson LO. 2000. Formation of harvest units with genetic algorithms. *For Ecol Manage.*
37
38 13 130(1): 57-67.
39
40
41
42 14 Lv Y, Duan Y, Kang W, Li Z, Wang FY. 2015. Traffic flow prediction with big data: a deep learning
43
44 15 approach. *IEEE Trans Intell Transp Syst*, Vol. 16, No.2: 865-873.
45
46
47
48 16 Monserud RA. 2003. Evaluating forest models in a sustainable forest management context. *FBMIS.*
49
50 17 1: 35-47.
51
52
53
54 18 Mäkelä A, Sievänen R. 1992. Height growth strategies in open-grown trees. *J Theor Biol.* 159(4):
55
56 19 443-467.
57
58
59
60

1
2
3
4 1 Mäkelä A. 1997. A carbon balance model of growth and self-pruning in trees based on structural
5
6 2 relationships. For Sci. 43(1): 7-24.
7
8
9
10 3 Mäkelä A. 2002. Derivation of stem taper from the pipe theory in a carbon balance framework. Tree
11
12 4 Physi. 22 (13): 891-905.
13
14
15
16 5 Mäkelä A. 2003. Process-based modeling of tree and stand growth: towards a hierarchical treatment
17
18 6 of multiscale processes. Can J For Res. 33: 398-409.
19
20
21
22 7 Mäkelä A, Mäkinen H. 2003. Generating 3D sawlogs with a process-based growth model. For Ecol
23
24 8 Manage. 184 (1): 337-354.
25
26
27
28 9 Mäkelä A, Pulkkinen M, Mäkinen H. 2016. Bridging empirical and carbon-balance based forest site
29
30 10 productivity - significance of below-ground allocation. For Ecol Manage. 372: 64-77.
31
32
33 11 Moazenzadeh R, Mohammadi B, Shamshirband S, Chau K. 2018. Coupling a firefly algorithm with
34
35 support vector regression to predict evaporation in northern Iran, Eng Appl Comp Fluid. 12
36 12 (1): 584-597.
37
38 13
39
40
41
42 14 Nelder JA, Mead R. 1965. A simplex method for function minimization. Comput J. 7 (4): 308-313.
43
44
45 15 Osyczka A. 1984. Multicriterion Optimization in Engineering with FORTRAN Programs.
46
47 16 Chichester: Ellis Horwood.
48
49
50
51 17 Pukkala T. 2009. Population-based methods in the optimization of stand management. Silva Fenn.
52
53 18 43(2): 261-274.
54
55
56
57 19 Pukkala T, Lähde E, Laiho O. 2010. Optimizing the structure and management of uneven-sized
58
59
60

stands of Finland. *Forestry*. 83 (2): 129-142.

Roise JP. 1986. A nonlinear programming approach to stand optimization. *For Sci*. 32 (3): 735-748.

Shinozaki K, Yoda K, Hozumi K, Kira T. 1964a. A quantitative analysis of plant form: The pipe model theory: I. Basic analyses. *Jpn J Ecol*. 14: 97-105.

Shinozaki K, Yoda K, Hozumi K, Kira T. 1964b. A quantitative analysis of plant form: The pipe model theory: II. Further evidence of the theory and its application in forest ecology. *Jpn J Ecol*. 14: 133-139.

Sofge D, Schultz A, De Jong K. 2002. Evolutionary computational approaches to solving the multiple traveling salesman problem using a neighborhood attractor schema. In: S. Cagnoni J. Gottlieb, E. Hart, M. Middendorf, G. R. Raidl, Editors. *Applications of Evolutionary Computing. EvoWorkshops 2002, LNCS 2279*. Springer-Verlag Berlin Heidelberg. p. 153-162.

Storn R, Price K. 1997. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim*. 11(4): 341-359.

Valsta L. 1990. A comparison of numerical methods for optimizing even aged stand management. *Can J For Res*. 20(7): 961-969.

Valsta L. 1992. A scenario approach to stochastic anticipatory optimization in stand management. *For Sci*. 38(2): 430-447.

Vesterstrøm J, Thomsen R. 2004. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *Proceedings*

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

of the IEEE Congress on Evolutionary Computation; Jun 9-23; Portland, OR. p. 1980-1987.

Wang PC, Shoup TE. 2011. Parameter sensitivity study of the Nelder–Mead simplex method. *Adv Eng Softw.* 42(7): 529-533.

Wikström P, Eriksson LO. 2000. Solving the stand management problem under biodiversity-related considerations. *For Ecol Manage.* 126(3): 361-376.

Zhou W. 1998. Optimal natural regeneration of Scots pine with seed trees. *J Envir Manage.* 53(3): 263-271.

Tables and Figures

Table 1. Initial stand states of seven Scots pine stands at CT (*Calluna*), MT (*Myrtillus*), and VT (*Vaccinium*) sites.

Table 2. Optimal rotations (yrs), number of thinnings (with dash) and objective function values (€ha⁻¹, in parentheses) for seven Scots pine stands by Osyczka's direct and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead(NM).

Table 3. Relative errors (%) of optimized and the highest objective function values for seven Scots pine stands by Osyczka's direct and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead(NM).

Table 4. Performance comparison of the bare land value (BLV, € /ha), the number of functional evaluations (NFE) and the amount of central processing unit time (CPU, seconds) required to converge for Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), and Nelder-Mead (NM).

Fig. 1. Basal area development by optimal solutions for stands 1-4 (a) and stands 5-7 (b).

Fig. 2. Thinning type at the 1st (a), 2nd (b), 3rd (c), 4th (d) thinning by ES and DRS for stand 6.

Fig. 3. Bare Land Value (BLV) as a function of Number of Functional Evaluations (NFE) by Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), and Nelder-Mead (NM) for stand 5.

Fig. 4. The minimum (dash-dot line), mean (solid line), maximum (dash line) objective function values by population-based algorithms. (a) Differential Evolution (DE), (b) Particle Swarm Optimization (PSO), (c) Evolution Strategy (ES), (d) Nelder-Mead (NM) for stand 5.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

1 **Fig. 5.** Comparison of robustness by Differential Evolution (DE), Particle Swarm Optimization
2 (PSO), Evolution Strategy (ES), Nelder-Mead (NM), Osyczka’s direct and random search algorithm
3 (DRS) for stand 5.

4 **Fig. 6.** Sensitivity of number of decision variables on objection function value by Osyczka’s direct
5 and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization
6 (PSO), Evolution Strategy (ES), Nelder-Mead (NM) for stand 6.

For Peer Review Only

Table 1. Initial stand states of seven Scots pine stands at CT (*Calluna*), MT (*Myrtillus*), and VT (*Vaccinium*) sites.

Stand	Age	#tree	BA	Hdom	H100	ST	TS
1	29	1500	6.4	7.9	17.3	CT	1300
2	23	1500	7.6	7.9	22.1	MT	1100
3	20	1500	7.4	8.0	28.3	MT	1300
4	27	1500	8.2	7.7	18.2	VT	1100
5	23	1500	7.0	7.9	25.7	VT	1300
6	23	2000	8.1	7.8	25.4	VT	1300
7	23	3000	10.0	7.8	24.8	VT	1300

Note: Age is initial age (yr), #tree denotes the number of trees per hectare, BA basal area ($\text{m}^2 \text{ha}^{-1}$), Hdom dominant height (m), H100 dominant height (m) at age 100, ST site type, TS temperature sum (d.d.).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Table 2. Optimal rotations (yrs), number of thinnings (with dash) and objective function values (€/ha, in parentheses) for seven Scots pine stands by Osyczka’s direct and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead(NM).

Stand	DRS	DE	PSO	ES	NM
1	99-3(883)	99-3(884)	99-3(884)	101-3(883)	98-3(880)
2	90-3(1557)	89-3(1557)	89-3(1558)	90-3(1555)	89-3(1556)
3	75-4(3718)	75-4(3715)	77-4(3729)	78-4(3713)	71-3(3699)
4	108-4(484)	109-4(485)	108-4(481)	105-4(483)	113-4(481)
5	84-3(2419)	84-3(2417)	84-3(2418)	86-3(2416)	85-3(2415)
6	92-4(2283)	98-4(2279)	87-4(2295)	83-4(2309)	91-4(2266)
7	96-4(2078)	93-4(2110)	89-4(2110)	96-4(2078)	91-4(2065)

Note: Bold font means the optimal solutions with the highest objective function values.

Table 3. Relative errors (%) of optimized and the highest objective function values for seven Scots pine stands by Osyczka's direct and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead(NM).

Stand	DRS	DE	PSO	ES	NM
1	0.11	0.00	0.00	0.11	0.45
2	0.06	0.06	0.00	0.19	0.13
3	0.29	0.38	0.00	0.43	0.80
4	0.21	0.00	0.82	0.41	0.82
5	0.00	0.08	0.04	0.12	0.17
6	1.13	1.30	0.61	0.00	1.86
7	1.52	0.00	0.00	1.52	2.13

1
2
3
4 1
5
6 2
7 3
8 4
9 4
10 5
11 6
12
13
14
15
16
17
18
19
20
21
22
23
24 7
25
26 8
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Table 4. Performance comparison of the bare land value (BLV, €/ha), the number of functional evaluations (NFE) and the amount of central processing unit time (CPU, seconds) required to converge for Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), and Nelder-Mead (NM).

n_{thin}	n_d	DE			PSO			ES			NM		
		BLV	NFE	CPU	BLV	NFE	CPU	BLV	NFE	CPU	BLV	NFE	CPU
1	5	2034	2526	353	2034	3131	438	2034	3101	434	2034	2178	304
2	9	2179	4546	636	2185	5781	809	2164	4141	579	2161	4469	625
3	13	2257	6566	919	2262	9231	1292	2253	8181	1145	2237	4697	657
4	17	2279	17086	2392	2295	26741	3743	2309	18221	2550	2260	4760	666
5	21	2299	21106	2954	2296	31581	4421	2300	18261	2556	2260	9019	1262
6	25	2298	25126	3517	2323	36421	5098	2299	18301	2562	2262	28025	3923

Note: n_{thin} denotes the number of thinnings, n_d the number of decision variables.

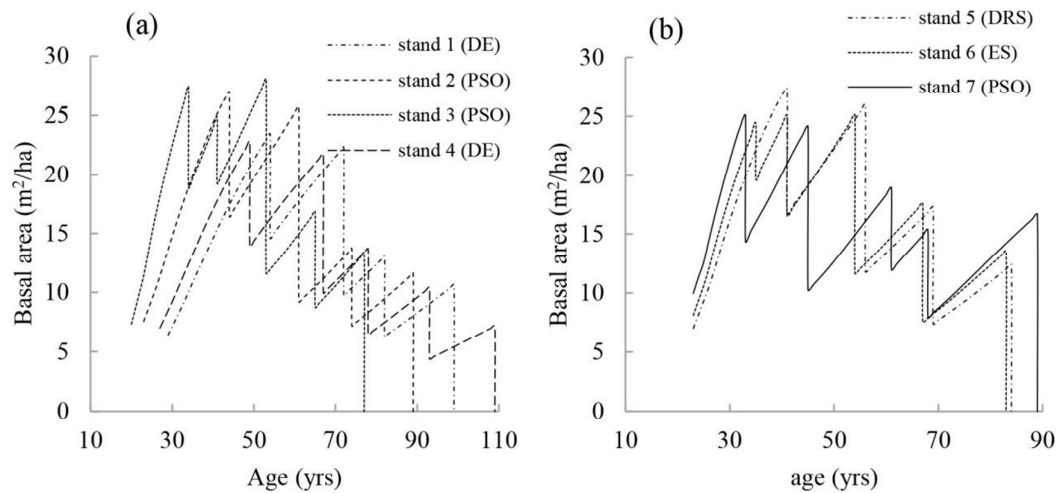


Fig. 1. Basal area development by optimal solutions for stands 1-4 (a) and stands 5-7 (b).

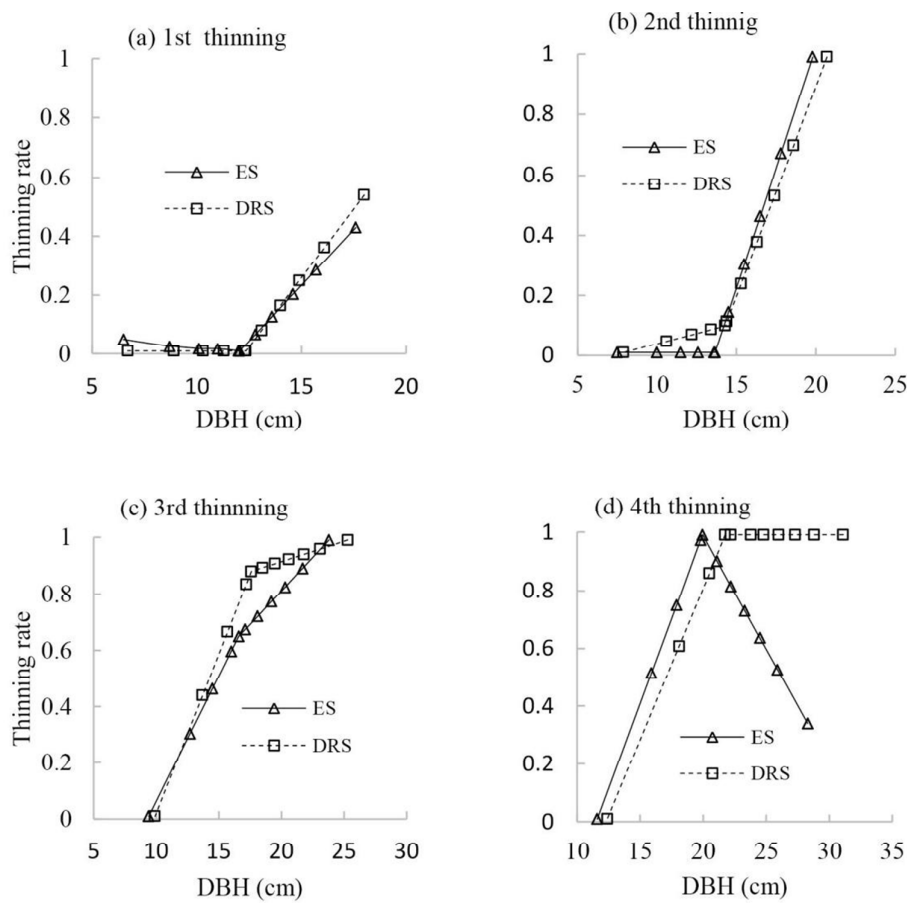


Fig. 2. Thinning type at the 1st (a), 2nd (b), 3rd (c), 4th (d) thinning by ES and DRS for stand 6.

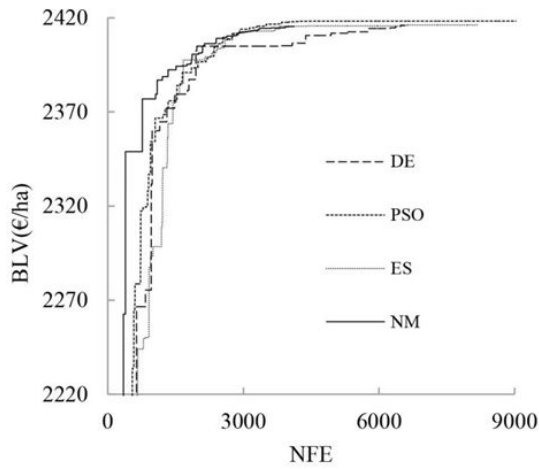


Fig. 3. Bare Land Value (BLV) as a function of Number of Functional Evaluations (NFE) by Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), and Nelder-Mead (NM) for stand 5.

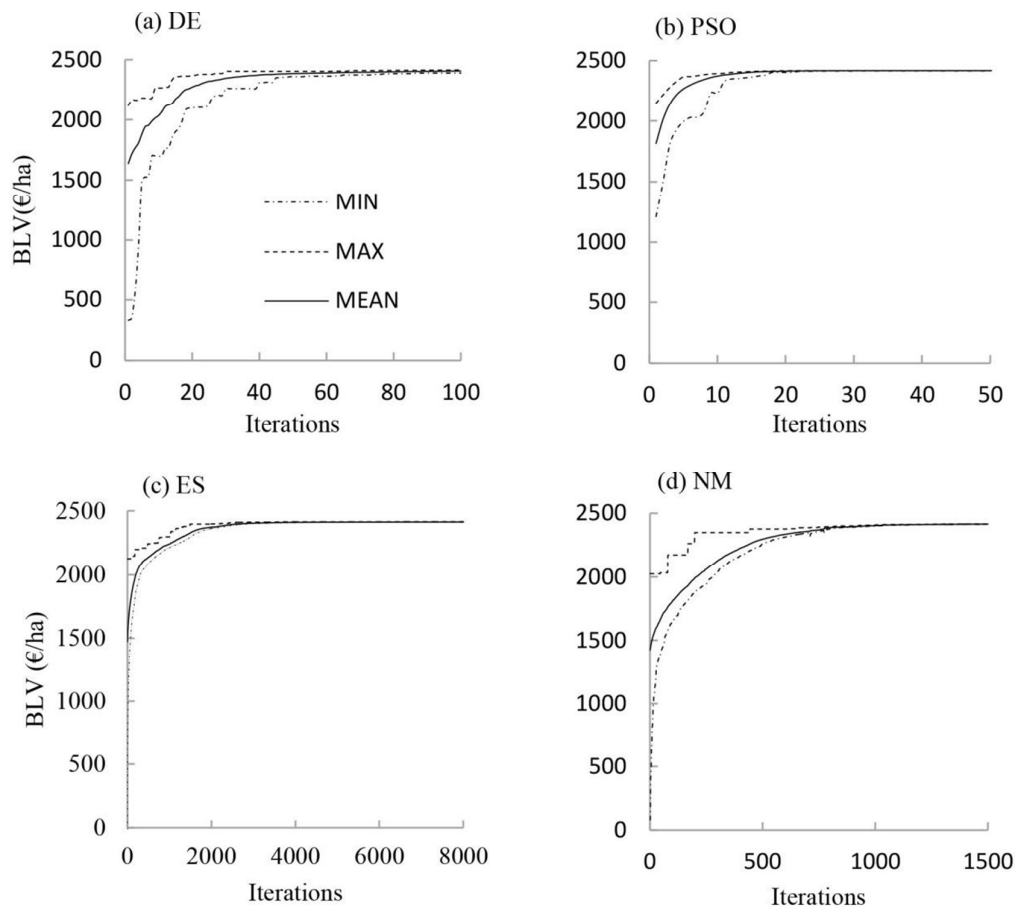


Fig. 4. The minimum (dash-dot line), mean (solid line), maximum (dash line) objective function values by population-based algorithms. (a) Differential Evolution (DE), (b) Particle Swarm Optimization (PSO), (c) Evolution Strategy (ES), (d) Nelder-Mead (NM) for stand 5.

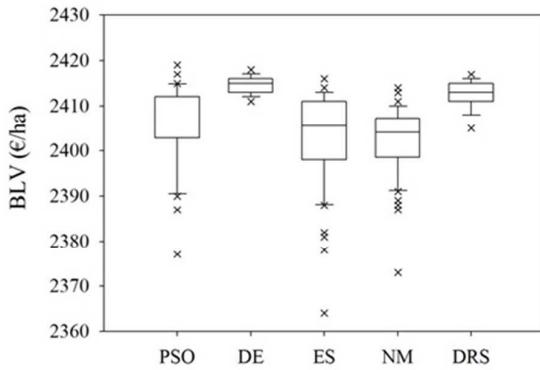


Fig. 5. Comparison of robustness by Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead (NM), Osyczka's direct and random search algorithm (DRS) for stand 5.

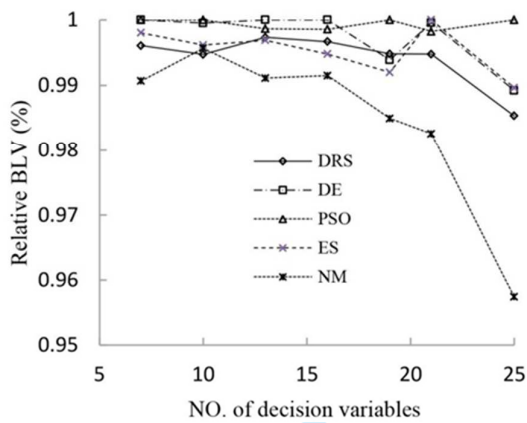


Fig. 6. Sensitivity of number of decision variables on objection function value by Osyxyzka’s direct and random search algorithm (DRS), Differential Evolution (DE), Particle Swarm Optimization (PSO), Evolution Strategy (ES), Nelder-Mead (NM) for stand 6.